A Machine Learning Approach to Detecting Depression in Social Media Users

Abhinav Sood

1. Abstract

Depression is a complex topic that can affect people due to a variety of factors. In this project specifically, we looked at whether or not we can detect depression in someone based on their posts on social media. Social Media continues to be a platform where people can speak their minds and express their feelings without the fear of communicating face-to-face with someone. Our goal with this project is to detect if someone is having depressing thoughts and potentially identify those who need help. To do this, we used the power of Sentiment Analysis and Natural Language Processing with the use of three different classifiers: Logistic Regression, Multilayer Perceptron Classifier and Random Forest Classifier. Our final model was a combination of the three that utilized the accuracy of each of three models. Our models will be able to predict with about 92% accuracy whether or not someone is having depressing thoughts through the input of their tweet.

2. Introduction

The detection of depression is vital in ensuring a healthy future for the youth of our world. Children and Young Adults struggle with the difficulties of everyday life and often find unhealthy ways of coping with their problems. By detecting potential depressing or harmful thoughts in someone early on could help save someone's life and help put them back on track. The use of machine learning to detect negative sentiments in social media is not a concept that is new, yet it continues to be one of the most effective ways of finding those that are in need of help.

3. Data

Our data comes from the *Sentiment Analysis for Tweets* on Kaggle which has aggregated 10,314 tweets for us to analyze consisting of 2,314 tweets that could be labeled as “depression” tweets and 8000 tweets that could be labeled as “non-depression” tweets. An example of a tweet labeled as a “depression” tweet is “I hate the bad days.. Been a long time since I've felt this low”. An example of a tweet that is labeled as a non depression tweet is “woo. its late! haha goodnight

twitterverse! Xoxo”. There are three columns in the dataset with the id of the tweet, the tweet itself, and whether or not the tweet can be labeled as someone with depression. The dataset set tweets with a value of 0 as non-depression tweets and tweets with a value of 1 as tweets posted by someone with depression. This dataset was not only diverse, but it was simple enough to work around. A limitation however is that the labeling of the tweets as “depression” or “no depression” is all subjective. A closer examination of the tweets however showed that these tweets' labels are valid and therefore serviceable. In addition to this, the dataset essentially included tweets that had the word “depression” in them. Therefore, a tweet that posts a link to an article titled “How to cure your depression” posted by CNN would be classified by the dataset as 1 or a depression tweet.

4. Related Works

Similar work has been done on detection of depression using Sentiment Analysis and Natural Language Processing. Student researchers at Stanford University, Diveesh Singh and Aileen Wang, utilized neural networks to detect depression in tweets. Their data was scraped from several twitter pages and was filtered to get rid of futile tweets.

5. Methodology

Before we started working on our models, we had to translate our data into something that our model could read. The inputs that we pass into our model are going to be tweets or rather strings. Our model has no way of deciphering what each word in the sentence means by itself. To allow our model to understand our sentences, we have to encode our sentences into vectors that can easily be inputted and read by our model. Using the Python framework Sentence Transformers, we were easily able to encode each of our sentences into a vector that our model will ultimately be able to read. Similar texts will have similar encodings.

Our next step is to define what are our inputs and what are our outputs. In this scenario, we are inputting vector representations of our tweets and we will be getting either a 0 for no depression and a 1 for someone who might have depression. Therefore, our “x” or our input are the messages and our “y” or our output is our depression result.

Now that our data has been encoded into something that our model can read into and we have clearly defined what our inputs and outputs are, we split our data into a roughly 67% training and 33% testing data split. Our training data will provide some guidance so that we can tweak our model, more specifically our weights so that our model is performing with the highest accuracy possible. Our weights values are initially randomized but through the process of training, we will

be able to find the optimal value for our weights. Having a large enough training dataset like the one we used prevents us from facing overfitting. Overfitting happens when the model becomes too comfortable with a particular dataset and ultimately throws off the results when other inputs are passed.

6. Testing

For testing, we used 3 different methods: Logistic Regression, MLP Classifier, Random Forest Classifier and Multiplicative Weight Update Method with 10,000 random tweets used.

The latter method is an algorithm that is based on the mathematical concept of game theory. Using the three classifiers that were implemented before, each classifier was assigned the same weightage initially. The model will first predict based on the majority choice of three models. In each successive case, the weights for each classifier are changed so that the more accurate the classifier is, the higher the weighting is. This ensures that our model is more biased towards the more accurate classifiers.

The results were as follows:

Classifier	Testing Accuracy	Training Accuracy
Logistic Regression	83%	99.92%
MLP Classifier	86.73%	99.98%
Random Forest Classifier	85.93%	99.98%
Majority choice of the Three	88%	99.98%
Multiplicative Weight Update Method	92%	99.98%

7. Evaluation

Most of our classifiers were about 99% accurate . Our most accurate classifier was the Logistic Regression model while our least accurate classifier was our Random Forest Classifier. This is a positive sign as our classifiers will be able to accurately detect whether or not a user has depression given their tweet with a very high accuracy. However there are some limitations. A lot

of the data used contained the word “depression” in their tweets. It is possible that our classifiers will start to look more for the word depression in tweets rather than look for words or phrases that are not as blatantly obvious when it comes to detecting depression.

8. Conclusion

The issue of depression is one that continues to affect people all over the world. People who are in serious need of help and do not realize it are letting the effects of depression control their life and their actions. The use of a model mentioned above could be vital in saving lives. Social media apps can implement this and have it running in the background. As a result, detection of depression can be a both efficient and streamlined process that could potentially save lives.

In the future, work will be done in order to make this model less susceptible to errors. Issues like sarcasm and poor word placement will be a focus moving forward in order to make the model as accurate as possible. The use of machine learning in detecting depression is one that should be researched further and could be a major step in helping those who are in need.

Sources

(Archived) Cs224n: Natural Language Processing with Deep Learning (Winter 2018), web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/. Accessed 15 Aug. 2023.

Ahmed, Arfan, et al. “Machine Learning Models to Detect Anxiety and Depression through Social Media: A Scoping Review.” *Computer Methods and Programs in Biomedicine Update*, 9 Sept. 2022, www.sciencedirect.com/science/article/pii/S2666990022000179.

Shinigami. “Sentimental Analysis for Tweets.” *Kaggle*, 3 May 2021, www.kaggle.com/datasets/gargmanas/sentimental-analysis-for-tweets.

Special thanks to Mr. Odysseas Drosis of Cornell University for his guidance with this project.